IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	)	
	)	
Guy L. STEELE, Jr.	)	Group Art Unit: 2193
	)	
Application No.: 10/035,579	)	Examiner: Ngo, Chuong D.
	)	
Filed: December 28, 2001	)	
	)	
For: FLOATING POINT SQUARE ROOT PROVIDER WITH EMBEDDED STATUS INFORMATION	)	Confirmation No.: 2901

Attention: Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

APPEAL BRIEF UNDER BOARD RULE § 41.37

In support of the Notice of Appeal filed September 30, 2005, and further to Board Rule 41.37, Appellant presents this brief and enclose herewith a check for the fee of $500.00 required under 37 C.F.R. § 1.17(c).

This Appeal Brief is being filed concurrently with a Petition for an Extension of Time for four months and the appropriate fee.

This Appeal responds to the Notice of Panel Decision from Pre-Appeal Brief Review mailed on November 7, 2005 and the Final Office Action mailed on May 25, 2005, which rejected claims 1-37 under 35 U.S.C. § 103(a).

If any additional fees are required or if the enclosed payment is insufficient, Appellant requests that the required fees be charged to Deposit Account No. 06-0916.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

Sun Microsystems, Inc. is the real party in interest, as indicated by the assignment in its name, recorded at Reel 012442, Frame 0992 on December 28, 2001.

II. RELATED APPEALS AND INTERFERENCES

In accordance with 37 C.F.R. § 41.37(c)(1)(ii), Appellant advises the Board of Patent Appeals and Interferences (the "Board") of the following pending appeals, which may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in the instant appeal:

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,747, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,595, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,584, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,587, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,647, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,580, in which an Appeal Brief was filed concurrently herewith.

III. STATUS OF CLAIMS

Claims 1-37 remain pending and under current examination.

Claims 1-37 have been finally rejected by the Examiner under 35 U.S.C. § 103(a) as unpatentable over U.S. Patent No. 6,009,551 to Lynch et al. ("*Lynch*"), and Appellant appeals the rejection of those claims. The attached Appendix contains a clean copy of the claims involved in the appeal, claims 1-37.

IV. STATUS OF AMENDMENTS

All amendments have been entered. No amendments under 37 C.F.R. § 1.116 have been filed.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claims 1, 14, and 26 of this application recite a system, method, and computer-readable medium for providing a floating point square root. *Specification*, p. 1, paragraph 002.

Digital electronic devices, such as digital computers, calculators and other devices, perform arithmetic calculations on values in integer, or "fixed point," format, in fractional, or "floating point" format, or both. *Specification*, p. 1, paragraph 003. Institute of Electrical and Electronic Engineers (IEEE) Standard 754, (hereinafter "IEEE Std. 754") published in 1985 and adopted by the American National Standards Institute (ANSI), defines several standard formats for expressing values in floating point format and a number of aspects regarding behavior of computation in connection therewith. *Id.*

In prior art devices that perform floating point computations, floating point status information generated by the computation is stored in a floating point status register. *Id.*

at p. 7, paragraph 020. The status information is stored as conditions, represented by flags that are stored in the floating point status register. *Id.* at p. 7, paragraph 026.

However, the modes (*e.g.*, the rounding modes and traps enabled/disabled mode), flags (*e.g.*, flags representing the status information), and traps that are required to implement IEEE Std. 754 introduce implicit serialization issues. *Id.* at p. 8, paragraph 028. Implicit serialization is essentially the need for serial control of access (read/write) to and from globally used registers, such as a floating point status register. *Id.* at p. 8, paragraph 028. The potential for implicit serialization makes the Standard difficult to implement coherently and efficiently in today's superscalar and parallel processing architectures without loss of performance. *Id.* at pp. 8-9, paragraph 028.

Moreover, the implicit side effects of a procedure that can change the flags or modes can make it very difficult for compilers to perform optimizations on floating point code. *Id.* at p. 9, paragraph 029. As a result, compilers for most languages usually assume that every procedure call is an optimization barrier in order to be safe. *Id.* This unfortunately may lead to further loss of performance. *Id.*

The claimed invention addresses these and other problems of prior art floating point computational systems. *Id.* at p. 9, paragraph 032. Since the floating point status information comprises part of the floating point representation of the result, instead of being separate and apart from the result as in prior art square root units, the implicit serialization that is required by maintaining the floating point status information separate and apart from the result may be obviated. *Id.* at p. 12, paragraph 045.

The invention, as recited by independent claim 1, relates to a system for providing a floating point square root (*Id.* at p. 10, paragraph 034). The system may

include an analyzer circuit (Fig. 1, 12) configured to determine a first status of a first floating point operand (Fig. 1, 11) based upon data within the first floating point operand (*Id.* at p. 18, paragraph 063; Fig. 2). Further, the system may include a results circuit (Fig. 1, 13, 14, and 15) coupled to the analyzer circuit (Fig. 1, 12) and configured to assert a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand (*Id.* at p. 18, paragraph 063; *Id.* at p. 13, paragraph 046 (Fig. 2); *Id.* at p. 29, paragraph 094 (Fig. 3); *Id.* at p. 33, paragraph 0121 (Fig. 4)).

The invention, as recited by independent claim 14, also relates to a method for providing a floating point square root (*Id.* at p. 10, paragraph 035). The method may include determining a first status of a first floating point operand based upon data within the first floating point operand (*Id.* at p. 18, paragraph 063; Fig. 2). Further, the method may include asserting a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand (*Id.* at p. 18, paragraph 063; *Id.* at p. 13, paragraph 046 (Fig. 2); *Id.* at p. 29, paragraph 094 (Fig. 3); *Id.* at p. 33, paragraph 0121 (Fig. 4)).

The invention, as recited by independent claim 26, further relates to a computer-readable medium on which is stored a set of instructions for providing a floating point square root, which when executed perform stages (*Id.* at p. 10, paragraph 036). The executed stages may include determining a first status of a first floating point operand based upon data within the first floating point operand (*Id.* at p. 18, paragraph 063; Fig. 2). The executed stages may also include asserting a resulting floating point operand containing the square root of the first floating point operand and a resulting status

embedded within the resulting floating point operand (*Id.* at p. 18, paragraph 063; *Id.* at p. 13, paragraph 046 (Fig. 2); *Id.* at p. 29, paragraph 094 (Fig. 3); *Id.* at p. 33, paragraph 0121 (Fig. 4)).

VI. GROUNDS OF REJECTION TO BE REVIEWED

A. Claims 1-37 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,009,511 to Lynch et al. ("*Lynch*"). *See Final Office Action mailed May 25, 2005*, pp. 2-3.

VII. ARGUMENT

A. Introduction

In view of the following arguments, Appellant respectfully requests the Board to reverse the Examiner's rejection of claims 1-37 under 35 U.S.C. § 103(a).

Several basic factual inquires must be made in order to determine the obviousness or non-obviousness of claims of a patent application under 35 U.S.C. § 103. These factual inquiries, set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 17, 148 USPQ 459, 467 (1966), require the Examiner to:

> (1) Determine the scope and content of the prior art;
> (2) Ascertain the differences between the prior art and the claims in issue;
> (3) Resolve the level of ordinary skill in the pertinent art; and
> (4) Evaluate evidence of secondary considerations.

The obviousness or nonobviousness of the claimed invention is then evaluated in view of the results of these inquiries. *Graham*, 383 U.S. at 17-18, 148 USPQ 467.

Thus, in order to carry the initial burden of establishing a *prima facie* case of obviousness that satisfies the *Graham* standard, the Examiner must show that the prior art reference teaches or suggests all the claim elements. *In re Royka*, 490 F.2d 981,

180 USPQ 580 (CCPA 1974). The Examiner must also show that there is some suggestion or motivation, either in the reference or in the knowledge generally available to one of ordinary skill in the art, to modify the reference. *In re Rouffet*, 149 F.3d 1350, 47 USPQ2d 1453 (Fed. Cir. 1998). "Even when obviousness is based on a single prior art reference, there must be a showing of a suggestion or motivation to modify the teachings of that reference." *In re Kotzab*, 217 F.3d 1365, 1370, 55 USPQ2d 1313, 1316-17 (Fed. Cir. 1998) (citations omitted). In addition, the suggestion or motivation "must be found in the prior art reference, not in the Applicant's disclosure." *In re Vaeck*, 947 F.2d 488, 493, 20 USPQ2d 1438, 1442 (Fed. Cir. 1991).

B. *Lynch* does not teach or suggest all of Appellant's claim elements.

The Examiner's rejections contain clear errors and omit essential elements necessary to establish a *prima facie* case of obviousness for Appellant's claims 1-37 based on *Lynch*.

Independent claim 1 recites:

> A system for providing a floating point square root, comprising:
> an analyzer circuit configured to determine a first status of a first floating point operand based upon data within the first floating point operand; and
> a results circuit coupled to the analyzer circuit and configured to assert a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand.

(emphasis added) as recited by claim 1.

The Examiner concedes that "Lynch et al. does not explicitly disclose the resulting status embedded with the resulting floating operand." *Office Action mailed May 25, 2005* at p. 2. No other reference is cited to cure this deficiency. Because

Lynch does not teach each and every element recited in the claims, no *prima facie* case of obviousness has been established.

Further, the rejection does not make clear what portion(s) of *Lynch* allegedly teaches the claimed "status." Even assuming that *Lynch's* tag value constitutes "status," (which Appellant does not concede) the tag value of *Lynch* is not "embedded within the resulting ... operand," as recited by claim 1.

The Examiner appears to assert that *Lynch's* element 84 (Fig. 4) constitutes an "operand" which contains tag field 89 (alleged status). *See Office Action mailed May 25, 2005* at p. 2. This is incorrect according to the disclosure of *Lynch*. Fig. 4 of *Lynch*, which Appellant reproduces below, clearly illustrates that Tag Field 89 (alleged status) and Reg Field 87 (operand) are separate from each other and are stored within register stack 84.

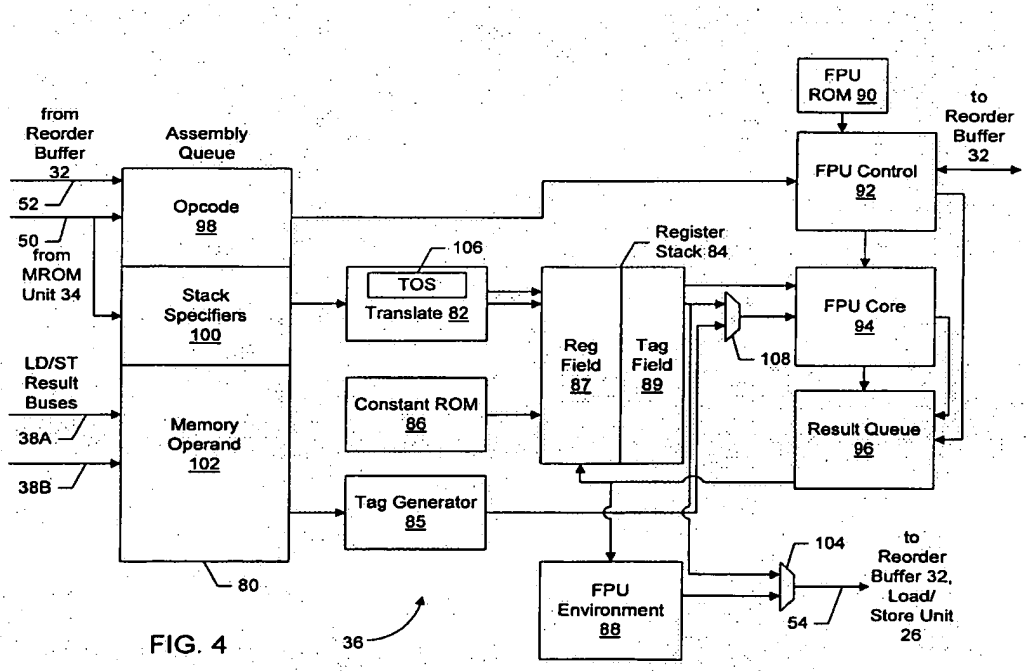

FIG. 4

Moreover, *Lynch* specifically states that element 84 is a register stack, not an operand, and that register stack 84 contains a separate Reg Field 87 for storing an operand and a separate Tag Field 89 for storing a tag (alleged status). *Lynch*, col. 15, lines 63-67. This does not constitute a teaching or suggestion of "a resulting status embedded within the resulting ... operand," as recited by independent claims 1, 14, and 26 and required by dependent claims 2-13, 15-25, and 27-37.

Because the *Lynch* does not teach or suggest "a resulting status embedded within the resulting ... operand," no *prima facie* case of obviousness has been established with respect to claim 1. Accordingly, the rejection of claim 1 under 35 U.S.C. § 103(a) based on *Lynch* is improper. Appellant requests the Board to allow this claim.

Moreover, claim 1 recites a combination including, for example a "resulting floating point operand containing the square root of the first floating point operand and a resulting status" (emphasis added). *Lynch* does not teach or suggest at least this additional element. Rather, as seen in Fig. 4 and discussed above, *Lynch* clearly illustrates that Reg Field 87 (the alleged operand) and Tag Field 89 (the alleged status) are separate. The separate tag value taught by *Lynch* is "appended to each floating point number." *Lynch*, col. 5, lines 44-45. Such teachings by *Lynch* do not constitute a teaching or suggestion of a resulting floating point operand containing both the "square root ... and a resulting status," as recited by claim 1 (emphasis added). Accordingly, the rejection of claim 1 under 35 U.S.C. § 103(a) based on *Lynch* is improper for at least this additional reason.

Because *Lynch* does not teach or suggest each and every element recited by claim 1, no *prima facia* case of obviousness has been established for this claim. Independent claims 14 and 26, although of different scope, recite elements similar to those recited by claim 1. Claims 2-13, 15-25, and 27-37 depend from independent claims 1, 14, and 25 and therefore include all of the elements recited therein. Accordingly, no *prima facie* case of obviousness has been established with respect to claims 2-37 for at least the reasons discussed above regarding claim 1. Appellant requests the Board to allow claims 1-37.

Moreover, dependent claims 2-13, 15-25, and 27-37 contain additional distinctions recited therein and are separately patentable. For example, the Examiner asserts for claims 2-4, 15-17, and 27-29 that "the use of buffers of storing operand and intermediate results in a processing circuit is well known" *Office Action mailed May 5, 2005* at p. 2. No reference is cited by the Examiner to support this assertion or to address any of the elements recited by claims 2-4, 15-17, and 27-29. The Examiner does not rely on *Lynch* for the elements of these claims, and Appellant treats this as a concession that *Lynch* does not teach or suggest such elements. Accordingly, for at least this additional reason, no *prima facia* case of obviousness has been established for claims 2-13, 15-25, and 27-37. Appellant requests the board to allow these claims.

C. There is no motivation to modify *Lynch*.

The threshold for establishing a motivation or suggestion to modify a prior art reference is high. The Federal Circuit has clearly stated that the evidence of a motivation or suggestion to modify a reference must be "clear and particular." *In re Dembicziak*, 175 F.3d 994, 999, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999). Further, the

Examiner can satisfy the burden of establishing a *prima facie* case of obviousness "only by showing some objective teaching in the prior art or that knowledge generally available to one of ordinary skill in the art would lead that individual to [modify or] combine the relevant teachings of the references." *In re Fine*, 837 F.2d 1071, 1074, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988) (citations omitted) (emphasis added). The Federal Circuit has reaffirmed the Examiner's high burden to establish a *prima facie* case of obviousness and has emphasized the requirement of specificity. *See Kotzab*, 217 F.3d at 1370, 55 USPQ2d, at 1317; *see also In re Sang-Su Lee*, 277 F.3d 1338, 61 USPQ2d 1430 (Fed. Cir. 2002). Moreover, "[e]ven when obviousness is based on a single prior art reference, there must be a showing of a suggestion or motivation to modify the teachings of that reference." *In re Kotzab*, 217 F.3d 1365, 1370, 55 USPQ2d 1313, 1316-17 (Fed. Cir. 1998) (citations omitted).

The Examiner relies on a single reference, *Lynch*, in rejecting claims 1-37. As addressed above, the Examiner concedes that *Lynch* fails to teach or suggest each and every claim element, nor is there any motivation or suggestion to modify *Lynch* to arrive at Appellant's claimed invention.

With respect to claim 1, the Examiner asserts that "since Lynch et al. disclose that each operand in the register stack (84) includes a value stored in a register field (78) [sic], and a tag stored in an associated tag field (89) ..., and a tag values is [sic] generated for each operation result ..., it would have been obvious to a person of ordinary skill in the art to store both the results with its tag in the register stack (84) in order to quickly determine its status in subsequence [sic] operations." *Office Action mailed May 25, 2005* at p. 3. This assertion by the Examiner fails to meet the requisite

high threshold of an objective indication that one of skill in the art would modify *Lynch* to arrive at Appellant's claimed invention for the reasons set forth below.

The Examiner's alleged motivation is directed to storing "both the result with its tag in the register stack." *Office Action mailed May 25, 2005* at p. 3 (emphasis added). This does not constitute motivation to provide "a resulting status embedded within the resulting ... operand," as recited by claim 1. For at least this reason, there is no motivation to modify *Lynch* to arrive at Appellant's invention as recited by claim 1.

Further, *Lynch* teaches an "apparatus and method for tagging floating point operands and results for rapid detection of special floating point numbers" (*Lynch*, title). *Lynch* is directed to "rapid" or "quick" detection of a status, yet, as conceded by the Examiner, *Lynch* does not teach or suggest the very element alleged to be obvious, namely a "resulting status embedded within the resulting floating point operand," as recited by claim 1. Thus, the Examiner's assertion that "it would have been obvious to ... store both the results with its tag in the register stack (84) in order to quickly determine its status in subsequence [sic] operations," even if true, both fails to cure the deficiencies of *Lynch* and further emphasizes that "a resulting floating point operand ... and a resulting status embedded within the resulting floating point operand" is not obvious to one of ordinary skill in the art. For at least this additional reason, there is no motivation to modify *Lynch* to arrive at Appellant's invention as recited by claim 1.

Indeed, Appellant's claimed invention distinguishes these teachings of *Lynch*. As set forth in Appellant's specification, "[s]ince the floating point status information comprises part of the floating point representation of the result, instead of being separate and apart from the result as in prior art square root units, the implicit

serialization that is required by maintaining the floating point status information separate and apart from the result can be obviated," *Specification* at p. 12, paragraph 045. That is, by providing "a resulting floating point operand ... and a resulting status embedded within the resulting floating point operand," as recited by claim 1, the problems of prior art systems that use a separate register field 87 and a separate tag field 89, as taught by *Lynch*, are addressed. For at least this additional reason, there is no motivation to modify *Lynch* to arrive at Appellant's invention as recited by claim 1.

Moreover, the Examiner has used impermissible hindsight in reconstructing Appellant's claimed invention. A determination of obviousness must "take into account only knowledge which was within the level of ordinary skill in the art at the time the claimed invention was made and ... [must] not include knowledge gleaned only from" Appellant's disclosure. *In re McLaughlin*, 443 F.2d 1392, 1395, 170 USPQ 209, 212 (CCPA 1971) (emphasis added). As discussed above, the Examiner concedes that *Lynch* does not teach "the resulting status embedded with the resulting floating operand." *Office Action mailed May 25, 2005* at p. 2. No other reference is cited to cure this deficiency. Therefore, the Examiner's determination of obviousness could not have been founded on knowledge within the level of skill in the art when Appellant's claimed invention was made. Such determination instead impermissibly relied on hindsight reasoning by looking to Appellant's specification. For at least this additional reason, there is no motivation to modify *Lynch* to arrive at Appellant's invention as recited by claim 1.

Because there is no motivation to modify *Lynch* to arrive at Appellant's invention recited by claim 1, no *prima facia* case of obviousness has been established for this

claim. Independent claims 14 and 26, although of different scope, recite elements similar to those recited by claim 1. Claims 2-13, 15-25, and 27-37 depend from independent claims 1, 14, and 25 and therefore include all of the elements recited therein. Accordingly, no *prima facie* case of obviousness has been established with respect to claims 2-37 for at least the reasons discussed above regarding claim 1. Appellant requests the Board to allow claims 1-37.

Moreover, the Examiner has not provided any motivation to modify *Lynch* to teach or suggest the elements recited by dependent claims 2-4, 15-17, and 27-29. As discussed above, the Examiner merely asserts that "the use of buffers of storing operand and intermediate results in a processing circuit is well known" *Office Action mailed May 5, 2005* at p. 2. This bare assertion does not provide the requisite motivation, nor does it meet the high threshold of an objective teaching that one of skill in the art would modify *Lynch* to arrive at Appellant's claimed invention. Accordingly, for at least this additional reason, no *prima facia* case of obviousness has been established for dependent claims 2-13, 15-25, and 27-37. Appellant requests the board to allow these claims.

D. Summary

The Examiner has not established a *prima facie* case of obviousness with respect to the presently appealed claims. In particular, the references cited by the Examiner in the appealed rejections do not teach or suggest all of the claim limitations, nor is there any suggestion or motivation, either in the cited references or in the knowledge generally available to one of ordinary skill in the art, to modify or combine

references. *See* M.P.E.P. § 2143. Thus, the Examiner has failed to meet the criteria required for a *prima facie* showing of obviousness.

VIII. CONCLUSION

For the reasons given above, the Examiner has not established a *prima facie* case of obviousness with respect to the appealed claims. Accordingly, pending claims 1-37 are allowable and reversal of the Examiner's rejections are respectfully requested.

To the extent any extension of time under 37 C.F.R. § 1.136 is required to obtain entry of this Appeal Brief, such extension is hereby respectfully requested. If there are any fees due under 37 C.F.R. §§ 1.16 or 1.17 which are not enclosed herewith, including any fees required for an extension of time under 37 C.F.R. § 1.136, please charge such fees to our Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: April 7, 2006

By: _____
Nathan A. Sloan
Reg. No. 56,249

IX. Claims Appendix to Appeal Brief Under Rule 41.37(c)(1)(viii)

1. A system for providing a floating point square root, comprising:

an analyzer circuit configured to determine a first status of a first floating point operand based upon data within the first floating point operand; and

a results circuit coupled to the analyzer circuit and configured to assert a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand.

2. The system for providing a floating point square root of claim 1, wherein the analyzer circuit further comprises:

a first operand buffer configured to store the first floating point operand; and

a first operand analysis circuit coupled to the first operand buffer, the first operand analysis circuit configured to generate a characteristic signal having information relating to the first status.

3. The system for providing a floating point square root of claim 2, wherein the first status is determined without regard to memory storage external to the first operand buffer.

4. The system for providing a floating point square root of claim 3, wherein the memory storage external to the first operand buffer is a floating point status register.

5. The system for providing a floating point square root of claim 1, wherein the results circuit further comprises:

a square root circuit coupled to the analyzer circuit, the square root circuit configured to produce the square root of the first floating point operand;

a square root logic circuit coupled to the analyzer circuit and configured to produce the resulting status based upon the first status; and

a result assembler coupled to the square root circuit and the square root logic circuit, the result assembler configured to assert the resulting floating point operand and embed the resulting status within the resulting floating point operand.

6. The system for providing a floating point square root of claim 5, wherein the square root logic circuit is organized according to the structure of a decision table.

7. The system for providing a floating point square root of claim 1, wherein the first status and the resulting status are each one of the following: an invalid operation status, an overflow status, an underflow status, a division by zero status, an infinity status, and an inexact status.

8. The system for providing a floating point square root of claim 7, wherein the overflow status represents one in a group of a +OV status and a -OV status.

9. The system for providing a floating point square root of claim 8, wherein the overflow status is represented as a predetermined non-infinity numerical value.

10. The system for providing a floating point square root of claim 7, wherein the underflow status represents one in a group of a +UN status and a -UN status.

11. The system for providing a floating point square root of claim 10, wherein the underflow status is represented as a predetermined non-zero numerical value.

12. The system for providing a floating point square root of claim 7, wherein the invalid status represents a not-a-number (NaN) status due to an invalid operation.

13. The system for providing a floating point square root of claim 7, wherein the infinity status represents one in a group of a positive infinity status and a negative infinity status.

14. A method for providing a floating point square root, comprising:

determining a first status of a first floating point operand based upon data within the first floating point operand; and

asserting a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand.

15. The method for providing a floating point square root of claim 14, wherein the determining stage further comprises:

storing the first floating point operand in a first operand buffer; and

generating a first characteristic signal representative of the first status.

16. The method for providing a floating point square root of claim 15, wherein the first status is determined without regard to memory storage external to the first operand buffer.

17. The method for providing a floating point square root of claim 16, wherein the memory storage external to the first operand buffer is a floating point status register.

18. The method for providing a floating point square root of claim 14, wherein the asserting stage further comprises:

producing the square root of the first floating point operand; and

asserting the resulting floating point operand.

19. The method for providing a floating point square root of claim 14, wherein at least one of the following: the first status, and the resulting status comprise at least one of the following: an invalid operation status, an overflow status, an underflow status, a division by zero status, an infinity status, and an inexact status.

20. The method for providing a floating point square root of claim 19, wherein the overflow status represents one in a group of a +OV status and a -OV status.

21. The method for providing a floating point square root of claim 20, wherein the overflow status is represented as a predetermined non-infinity numerical value.

22. The method for providing a floating point square root of claim 19, wherein the underflow status represents one in a group of a +UN status and a -UN status.

23. The method for providing a floating point square root of claim 22, wherein the underflow status is represented as a predetermined non-zero numerical value.

24. The method for providing a floating point square root of claim 19, wherein the invalid status represents a not-a-number (NaN) status due to an invalid operation.

25. The method for providing a floating point square root of claim 19, wherein the infinity status represents one in a group of a positive infinity status and a negative infinity status.

26. A computer-readable medium on which is stored a set of instructions for providing a floating point square root, which when executed perform stages comprising:

determining a first status of a first floating point operand based upon data within the first floating point operand; and

asserting a resulting floating point operand containing the square root of the first floating point operand and a resulting status embedded within the resulting floating point operand.

27. The computer-readable medium of claim 26, wherein the determining stage further comprises:

storing the first floating point operand in a first operand buffer; and

generating a second characteristic signal representative of the first status.

28. The computer-readable medium of claim 27, wherein the first status is determined without regard to memory storage external to the first operand buffer.

29. The computer-readable medium of claim 28, wherein the memory storage external to the first operand buffer is a floating point status register.

30. The computer-readable medium of claim 26, wherein the asserting stage further comprises:

producing the square root of the first floating point operand; and

asserting the resulting floating point operand.

31. The computer-readable medium of claim 26, wherein at least one of the following: the first status, and the resulting status comprise at least one of the following:

an invalid operation status, an overflow status, an underflow status, a division by zero status, an infinity status, and an inexact status.

32. The computer-readable medium of claim 31, wherein the overflow status represents one in a group of a +OV status and a -OV status.

33. The computer-readable medium of claim 32, wherein the overflow status is represented as a predetermined non-infinity numerical value.

34. The computer-readable medium of claim 31, wherein the underflow status represents one in a group of a +UN status and a -UN status.

35. The computer-readable medium of claim 34, wherein the underflow status is represented as a predetermined non-zero numerical value.

36. The computer-readable medium of claim 31, wherein the invalid status represents a not-a-number (NaN) status due to an invalid operation.

37. The computer-readable medium of claim 31, wherein the infinity status represents one in a group of a positive infinity status and a negative infinity status.

X. Evidence Appendix to Appeal Brief Under Rule 41.37(c)(1)(ix)

Appellant relies on Fig. 4 of *Lynch*, which is reproduced below.

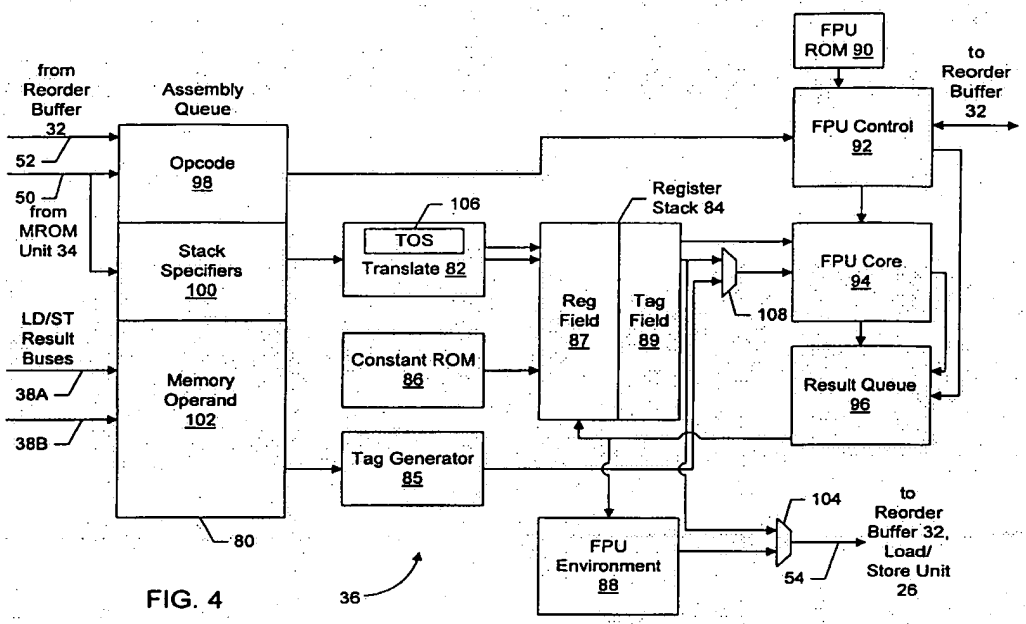

FIG. 4

U.S. Patent Dec. 28, 1999 Sheet 4 of 8 6,009,511

XI. Related Proceedings Appendix to Appeal Brief Under Rule 41.37(c)(1)(x)

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,747, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,595, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,584, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,587, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,647, in which an Appeal Brief was filed concurrently herewith.

The Appeal of In re Application of Guy STEELE, Jr., U.S. Patent Application No. 10/035,580, in which an Appeal Brief was filed on concurrently herewith.